

A SECOND ISSUE - AND FOR MMSFORTH, A SECOND YEAR! (Editorial)

If you've been enjoying MMSFORTH and its Newsletter half as much as we've been enjoying getting them to you, here are some ideas for spreading the fun. MMS hopes to keep its prices down; to do so, we must keep MMSFORTH sales volume up or lose the profit which guarantees next year's support. Also, a good thing like this deserves more visibility so you can hear your cluhs, friends, and those magazine articles talking up something more advanced than BASIC, at least part of the time. Perhaps you've noticed the silence?

Here's how you can help. Demo MMSFORTH to your friends and to your TRS-80 user group. Ask MMS or your local dealer for an appropriate number of free MMSFORTH fliers and hand them out to those who are interested. When the spirit moves you, send a Letter to the Editor to your favorite magazine(s). For a larger project, submit an article and include a short MMSFORTH program of your own as an illustration - one to create a table of DECIMAL to HEX conversions, for example, would be good. This sort of activity will get you famous and might earn you \$100, too! The magazines would like these articles and smart readers trust user letters more than company advertisements. MMS will be happy to answer your questions and, if you wish, to review your draft article for accuracy and newest information. If you don't speak up, who will? If lots of you speak up, MMSFORTH and your use of it will prosper!

Since completing MMSFORTH V1.9 (more on that later), MMS has been busy hatching a series of custom modifications of THE DATAHANDLER V1.1 for clients - who all are quite pleased with the results. One of these had relegated bis TRS-80 to book-end status after several unsatisfactory starts with Radio Shack and independent programmers - now at last he is seeing daily profits from its use. We're pleased, too. His DATAHANDLER-based inventory program is outperforming many we have seen on minicomputers and any other we've seen for the micros! Customizing THE DATAHANDLER is fun, and can provide a comfortable home computer income to an experienced user of MMSFORTH. Perhaps you.

Good feedback from MMSFORTH users has improved our MMSFORTH V1.9 User Instructions. A section is reprinted herein, entitled HOME REMEDIES. Enjoy!

-- Dick Miller, Editor 4th Class

MMSFORTH V1.9 HAS NEW FEATURES:

MMSFORTH improvements keep coming! Unlike most software products, we offer them to present users as easy upgrades as soon as they come available. We are celebrating the first birthday of MMSFORTH with another release of the MMSFORTH System, as well as a major lower-case re-write of Part 1 of the MMSFORTH Users Instructions and a minor revision of Part 2, the Glossary. Part 1 and Part 2 are available as upgrades at \$4.00 each plus \$1.00 shipping/handling.

So far, Version 1.9 is available only for disk. Like prior versions, there are two ways you can step up to it for \$10.00 plus \$1.00 shipping/handling. Send your original MMS-labelled MMSFORTH System Diskette for a re-write at MMS. (Include \$4.00 extra, which will be returned unless we have to supply a fresh diskette.) Or you can order an incomplete MMSFORTH V1.9 Upgrade Diskette for merging by yourself. Either way, you'll get your money's worth: even if you have V1.8, V1.9 adds such goodies as upper-lower case characters (just toggle your Radio Shack-modified keyboard, etc., on or off with a shift-0), a full-ASCII keyboard (to output to the display, printer, or MODEM all those characters Radio Shack forgot to provide), and a CUSTOMIZE command (for painless adjustments of MMSFORTH to your TRS-80 hardware and easy merges onto non-system diskettes). The icing on the cake is a marvelous sixth demonstration program: BREAKFORTH with sound, by Arnold Schaeffer! This real-time graphics game is sure to fascinate you and your friends, while showing the modular programming and lightning speed capabilities of high-level MMSFORTH.

Version 1.9 also changes a few internals. The MMSFORTH full-screen editor has been so well accepted that we no longer consider it optional. It is ahoard from boot; the E command for one-line editing has been dropped to maintain RAM economy and because it now is not needed. New words include ENTER (use it to print " (ENTER)" and pause), OR (a new friend of AND, XOR, etc.), and ALLOT (defined as H + !). We'll be using them later. And without external change, we have recoded <BUILDS and DOES> for still greater efficiency.

How often should you upgrade? The decision is your own, and MMS makes it easy to afford when you are ready. Please remember that each copy is just for its registered owner, and that we want you to be able to take pride in our latest work - as we do!

A MERGE COMMAND FOR THE DATAHANDLER:

You're in for a treat when you meet DATAHANDLER Version 1.1, the latest upgrade to our database management system in MMSFORTH. It does almost everything remarkably better than other database systems on microcomputers. In part, this is accomplished by trading off giant size files capabilities for extreme speed and flexibility. Files larger than memory are still useable once split into subfiles.

Here is an add-on MERGE routine for THE DATAHANDLER which will enable you to re-shuffle subfiles and then to merge the records from two for a new use. The routine only checks that the two files being merged have the same number of fields. The first file is loaded using a standard GET command, and then the files you want merged are accessed one at a time using the MERGE command. The resulting file has the descriptor and fields of the first file. This file is stored using the SAVE command.

You also may use the MERGE command to change the descriptor and field names of an existing file. To accomplish this, create a new file with the required descriptor and field definitions; then MERGE the old file's records into this empty file.

Be sure to leave RAM space for the MERGE command's compiled code. This space can be obtained by setting M#BLKS (in Block 40) to 1 less than the usual suggested kilobytes of filespace: 8 for a 32K RAM system and 24 for a 48K RAM system. Or, you can delete commands you do not need (LABELS, etc.) to provide room to have the MERGE command permanently loaded.

BLOCK : 82

```
0 ( MERGE NEW DATA TO CURRENT DATAHANDLER FILE, TD/MMS, 6/80 )
1 0 CONSTANT BGAD 0 VARIABLE WHERE
2 : GNXTBLK NXTBLK @ IF NXTBLK @ DBLK + BLOCK DUP (') BGAD !
3 NXTBLK @ LINK @ NXTBLK ! ELSE 0 THEN ;
4
5 : ?NXT WHERE @ DUP 1023 > IF 1024 - GNXTBLK DROP THEN
6 DUP 1+ WHERE ! BGAD + C@ ;
7
8 CODE HI HL POP L H MOV 0 L MVI PSH
9
10 : MERGE CLS WHILE G?N -1 = PERFORM PEND
11 GNXTBLK 0= BGAD @ 0= + IF " FILE EMPTY. " QUIT THEN
12 BGAD 2 + @ FIELD-# @ <> IF " FILES NOT COMPATIBLE. " QUIT THEN
13 BGAD 10 + @ BGAD 18 + @ - WHERE ! BGAD @ 0 DO
14 PAD 2+ ?NXT ?NXT HI + DUP PAD ! 2 - 0 DO ?NXT OVER C! 1+ LOOP
15 DROP +RECORD +INDEX LOOP CR MEMORY ;
```

SUBSCRIBE NOW TO CONTINUE RECEIVING THIS NEWSLETTER!

MMS has delivered the first two issues of the MMSFORTH Newsletter free, as promised. Your feedback says that the small print size is acceptable (and it allows us to cram 14 single-spaced pages down to 4!). The Fourth-Class mailing time is too slow, so future issues will go First Class. We can provide another four issues of similar size to our MMSFORTH licensees for your \$10.00 check to MMS. We phrase it this way to allow flexibility: if the amount of information escalates, we will deliver at least this much text for the price; if we miss deadlines, we will still deliver the same amount before setting the renewal time.

Will four MMSFORTH Newsletters be worth \$10.00 to you? Although our small number of licensees compared to magazine readerships makes our relative costs high, we feel confident that any active user of MMSFORTH will more than get the money's worth from even a single issue. Upcoming articles will bring you a free Absolute Assembler, to permit the creation of standard SYSTEM load modules with Forth's advanced assembler capabilities. More news, fixes, routines, insights and fun for beginners and advanced Forth users alike. More tools for hitching THE DATAHANDLER to your tough jobs. And more of your own user questions and suggestions. If you haven't yet warmed up to Forth, the MMSFORTH Newsletter may help. If you have, it's sure to be a hit!

Copy the attached form to get aboard. Each registered MMSFORTH licensee is entitled to order a single subscription. Please ACT IMMEDIATELY! Arrival at MMS by July 15th assures prompt mailing of your July-August issue.

Mail to: MMSFORTH Newsletter Subscription (Issue No's. 3-6)
MILLER MICROCOMPUTER SERVICES
61 Lake Shore Road, Natick, MA 01760

Name: MMSFORTH S/N:

Company:

Address:

City, State, Zip:

MY \$10.00 CHECK TO MMS IS ENCLOSED. Check No. is:

SHIPPING -- A VOYAGE INTO FORTH

Many beginning Forth programmers dislike the typical Forth article's emphasis on high-level math or machine-code detail. Here is an introduction to MMSFORTH which flexes some of its muscles in just plain fun -- sailing a fleet of ships across your TRS-80's video display!

Because Forth is structured and has an extensible dictionary of instructions (words), we can build our "program" (actually, just a temporary extension to the dictionary) by simultaneous top-down structuring and bottom-up programming. Let's try this new and powerful approach. Our SHIPPING program will SAIL a BOAT across the screen 5 times. So, taking it from the top, the final word we will define will be SHIPPING:

```
: SHIPPING 5 0 DO SAIL LOOP ;
(The 5 0 DO ----- LOOP is equivalent to BASIC's "FOR I=1 TO 5
-----: NEXT I".)
```

First load basic Forth into your TRS-80. This takes under 4 minutes for the 10.5-K System Cassette, or just 4 seconds to "boot up" the System Diskette. Now we must find a suitable block number at which to develop our program. If you just received your MMSFORTH System Diskette, boot it and say DIR to display its directory menu. Format a blank disk with the FMT command or just use any existing formatted disk. Use BACKUP or OBACKUP to copy the original to Drive 1 or Drive 0, respectively. Now store your original MMSFORTH Diskette back in safekeeping and boot the new copy. MMSFORTH tape users can parallel these activities by loading basic Forth (and Block 65 if prior to V1.9), then dumping it to a fresh tape with the FORTH-DUMP command. (See the MMSFORTH Users Instructions for further tape operation variations.)

What block number will we use? Check the availability of unused blocks by INDEXING your disk. For instance, 60 86 INDEX displays the 0-Line of each upper-range block on Drive 0. The 0-Line is ordinarily used to label the block (like a BASIC remark, in parentheses). If Block 79, for example, looks available and desirable, confirm that it is empty by reading that block (copying its contents) to the video RAM: 15360 79 RELK . An empty block, formatted with HEX E5's, looks like a checkerboard pattern. If other data exists, find a better block! Next, CLEAR it to blanks with: EDITOR 79 CLEAR . (EDITOR appends the EDITOR vocabulary, as does LIST or EDS.)

Now put a title on your new block's 0-Line: 0 P (79 - SHIPPING DEMO FROM MMSFORTH NL NO. 1:2) : TASK ; TASK is a dummy definition and will be explained later. If no corrections are needed, enter to substitute your new line for the prior contents of Block 79, Line 0. Let's proceed to draw a BOAT using the regular TRS-80 graphics characters on a single line. Let's first define a word GRAPHICS to tabulate the 64 ASCII Codes beginning at 128, and their resulting graphics characters: : GRAPHICS CR 64 0 DO I 128 + DUP 6 .R ECHO LOOP CR ; Put this GRAPHICS definition on Line 2 by preceding it with "2 P " (do not enter) and immediately keying it in. (There is no specific need to skip Line 1; it's just a good habit to leave some space for future adjustments.)

To try running this much, enter "79 LOAD". As the block is loaded, in turn each immediate command is executed and each definition is compiled into fast, compact machine code in RAM. Now enter GRAPHICS to execute our newest Forth word.

Any errors? Just enter V to re-View the same block, then use the down-arrow to move the cursor to Line 2. You are ready to edit with submodes D for Delete, R for Replace, or I for Insert. (Then S to Substitute if okay, or Q to Quit if not.)

Now we can use our new table like an artist's palette to create our BOAT. You may prefer a different design, but here's the one I chose. Starting with its stern (back) at the left and moving to the bow, I jotted down my selection:

```
172 188 176 186 176 176 188 190 135 . Then I reversed the
whole set, because the first one drawn will be the top one on the
stack, or the last one in our definition. ("LIFO" -- last in,
first out.) As an opening task, I remembered to add a first blank
graphic character (ASCII 128, last in my definition) so each
successive picture, shifted one space to the right, will overwrite
all characters in the prior one and thus erase the trail that
otherwise would be left behind. Once the ASCII Codes are set on
stack, we just follow with an equal number of ECHO commands to
output them to screen.
```

```
Now the whole definition of BOAT can be put on Lines 3 and 4:
: BOAT 135 190 188 176 176 186 176 188 172 128
10 0 DO ECHO LOOP ;
```

We'd like to try this new word now, but loading the revised block would add a second compilation of GRAPHICS, too. This is why we inserted Forth's dummy word TASK at the beginning of the operation. "FORGET TASK 79 LOAD" clears all the recently-compiled code from RAM and then re-compiles. Now enter BOAT. Did you do it right? A lot faster and more compact than BASIC, isn't it!

Now to SAIL the BOAT once across the screen. Where will we start drawing the boat? I chose the left-most position on the 5th line. One-upping Level II BASIC's PRINT @ statement, we use MMSFORTH's PTC command to put the cursor at Y down and X across: : SAIL 55 0 DO 5 I PTC BOAT LOOP 5 I 10 - PTC 30 ECHO ;

When SAIL is called, the BOAT will be drawn and redrawn at positions incremented from 0 to 54. (54 was determined to be the rightmost position without wraparound.) At the end of the line, the cursor is immediately backed 10 spaces and the final BOAT picture is erased with a 30 ECHO (equivalent to BASIC's PRINT CHR\$(30), which clears to end of line). Enter this SAIL definition on Line 5. Then "FORGET TASK 79 LOAD" to see how it runs.

After viewing this, I thought the display would be more effective if I slowed it down a bit. So in the definition of SAIL I inserted the words "100 PAUSE" just ahead of LOOP. You can do this too: move the cursor down to Line 5, then across to LOOP, and "I" to insert PAUSE. Then define PAUSE on Line 7:

```
: PAUSE 0 DO LOOP ;
This is equivalent to a BASIC "FOR I=0 TO --- : NEXT"
statement. The upper limit is as yet undefined, but that is the
100 preceding PAUSE in the SAIL definition line. This nifty
structure is one bonus of Forth's Reverse Polish Notation and its
modular design. Note that PAUSE will be used here as a nested
loop.
```

SAIL cannot compile unless PAUSE is previously defined, so we have to shift the PAUSE definition ahead of SAIL in the block. Do this by placing the cursor on Line 7, shift-D, moving the cursor to Line 5, and shift-I. This deletes Line 7 from the block buffer but stores it in PAD, then inserts PAD before Line 5 (i.e., as the new Line 5), shifting Lines 5 and 6 up to 6 and 7. (In MMSFORTH's LIST editing mode and in less sophisticated Forth editors, the line-insert command inserts after the line, not before. MMSFORTH's Full-screen Editor does it differently in order to insert ahead of Line 0. The LIST editing mode can insert ahead of Line 0 with the words, " -1 I".)

As before, you can clear, recompile and test this change. I edited further to try other PAUSE times, finally settling on "200 PAUSE" as my choice.

Next, we define SHIPPING to SAIL 5 times and then print "ENTER '79 LOAD' FOR SHIPPING.". A final CR keeps the completion "OK" from confusing the end of this text line.

So much for the definitions. During the loading of Block 79, we can follow up on the compilation of these definitions with an immediate command on its final line: "SHIPPING FORGET TASK". Now the action takes place, the instruction for repeat is displayed, and the RAM is cleared for another run.

Go Forth to happy sailing!

BLOCK : 79

```
0 ( SHIPPING DEMO, MMS/ARM, 8/4/79 ) : TASK ;
1
2 : GRAPHICS CR 64 0 DO I 128 + DUP 6 .R SPACE ECHO LOOP CR ;
3 : BOAT 135 190 188 176 176 186 176 188 172 128
4 10 0 DO ECHO LOOP ;
5 : PAUSE 0 DO LOOP ;
6 : SAIL 55 0 DO 5 I PTC BOAT 200 PAUSE LOOP
7 5 I 10 - PTC 30 ECHO ;
8 : SHIPPING 5 0 DO SAIL LOOP
9 10 16 PTC " ENTER '79 LOAD' FOR SHIPPING." CR ;
10
11
12
13
14 SHIPPING FORGET TASK
15
```

SORT DEMO:

Some new users have requested an explanation of the SORT program on our MMSFORTH System Diskette or Cassette. This program allows you to select from a series of sorting algorithms, then to enter a number from 1 to 1023. The program generates that many random numbers of ASCII code, placing their number, letter, symbol and graphic characters side by side on the CRT. As soon as they are generated, the program uses the sort routine you chose to sort these characters into ASCII-code order, right in place on the screen and in real time!

The first four sorts are all in high-level Forth, while AQUICK is exactly identical to QUICK except for a six-line recoding in 8080 Assembler of QUICK's three-line PARTITION routine. With that change, a 1,000-item sort takes about 3 seconds instead of 8 - which wasn't bad, either! We recommend short sorts for the slower methods: 128 or 256 characters for INSERT and SELECT. (We didn't try a bubble sort - they're too slow to compete except as a beginner's exercise. However, if someone wishes to provide this code for comparison purposes, we'll be pleased to run it in this Newsletter.) SHELL and QUICK are fast enough to permit quick full-screen sorts, so after you try short ones to compare with the first two, try 896 (14 lines) or 1023 to see them do some real work! This unusual demonstration of sorting techniques should be more than enough to convince any doubter of the speed and usefulness of Forth as a serious language.

SYSTEM LOCK-UP:

Fortb is a relatively forgiving language with simple and effective error comments. However, one of the more puzzling introductory aspects of Fortb is the ability to "lock up" the entire system with no cursor. In this event, the disk system programmer can simply reboot; in a few seconds the system will be back to debug his/her program or to avoid the improper entry. But sometimes the disk user will have a large and slow-loading program, and the tape user won't want to wait the four minutes to reboot.

USUALLY THE FORTH PROGRAM IS STILL IN RAM AND THERE IS A DIRECT WAY TO RECOVER IT. Boot your TRS-80 into Level II BASIC. (If you are using an Expansion Interface Unit, be sure to hold down the Break key while pressing the Reset button.) Press Enter, then enter SYSTEM, finally enter /19200 (that's the entry point of MMSFORTH). If you're lucky, you're back in business! With an Expansion Interface, the first lines of the current block will be corrupted; an ERASE-CORE will ignore it.)

The ability to lock up the system is not an accident of Fortb design. Rather, it is the result of giving the programmer unusual access to the computer itself - one of the strongest features of Fortb. The programmer is left to provide further limitations when desired for a particular task. In the absence of limitations, the lock-up can occur in several ways. Usual causes are "blowing the stack", or asking for something you can't have but which you haven't instructed the computer to reject.

Example (Bad Request):

Your MMSFORTH System is set for two disk drives, but you don't have a formatted diskette in Drive 1. You select Block 50 and hang the system. What's wrong?

Most probably you were in HEX (Base 16) rather than DECIMAL mode. Use the recovery procedure described above. Then try again. Just before you select Block 50, confirm that you haven't left the system in HEX instead of DECIMAL!

Example (Blowing the Stack):

You have just LOADED the appropriate blocks to run the SORT demo on your MMSFORTH System Tape. You pressed Break, tried a Fortb routine from screen, then went to reLOAD the SORT blocks but hung the system. What happened?

If you had exited the SORT program by entering STOP from its menu screen, it would have executed a FORGET TASK to remove the SORT code from the Fortb Dictionary. After pressing the Break key you could have done the same. But instead you left this code, which nearly fills available space in your 16K RAM; then you attempted to reLOAD. Wrong! To LOAD is to compile into Dictionary, and no way do you have room for a second complete set of SORT words in your tightly-packed RAM! The LOAD proceeds and

BREAKFORTH:

If you liked Digi-Pong, wait til you meet BREAKFORTH! This enticing game is a new add-on to the Version 1.9 MMSFORTH System Diskette. It also will be featured in an article in the August 1980 SPECIAL FORTH ISSUE (hear, hear!) of BYTE Magazine. Arnold Schaeffer, a high-school student in New Hyde Park, N.Y., used the MMSFORTH Tape Cassette to create it as his first project in Fortb! We were so pleased with his tape that we bought the rights and will be using it as a MMSFORTH tutorial. To this end, we have expanded it from Arnold's three packed blocks (like the ones we cram aboard on our MMSFORTH System) to six relatively pretty ones (like the ones we write for development projects and you should write for yourself!). We'll examine it in a future Newsletter issue. For now, we just want to get you started enjoying it.

To play BREAKFORTH, call it from Directory or load its six blocks of source code directly. Select ball speed and number of balls; you might start with short games using speed of 7 and 5 balls, move on to a more challenging speed of 4 and 50 balls for high scores. Once you enter the number of balls, the game begins. Use the right and left direction arrow keys to move the paddle, and good luck! You will discover that bricks score more than 1 point each as you remove them from higher levels, ball speed also picks up as you get higher, and the paddle has some tricky backspin! We've added a BEST SCORE counter, so you can track your improvement. If you are one of those people who NEVER quit once starting a game, know that you can press the shift-@ keys to stall while you take a break. Or, to start again while maintaining the BEST score, press Break, key in BREAKFORTH, and press Enter.

To add sound, just plug an external speaker into the EAR jack of your cassette recorder, attach the middle cable from the keyboard unit (but not the motor remote cable), and open the tape compartment door. While depressing the Record-interlock finger at the left side of the back of this compartment, simultaneously press the Record and Play keys. The tape recorder will presume it is monitoring a recording, while you hear the bouncing ball!

Oh yes, you also get a giant bonus once you remove ALL the bricks! Have fun using MMSFORTH's birthday gift from Arnold Schaeffer!

the Dictionary grows up into higher RAM, until it eats up the last remaining bytes where the User Stack is growing down and chews on into it. The stack is blown and, on the next stack operation, so is the system.

You may be able to recover, FORGET TASK, and run the underlying program. Or you may have to start from scratch.

AMNESIA:

As our term suggests, in this case the system doesn't remember much of anything. You may be in the wrong Vocabulary - enter FORTH and try again. If the system returns FORTH? you have blown the Dictionary. Congratulations! There is no recovery. You have messed up the address pointers which are essential to Fortb's indirect threaded code (ITC) and the lower words can no longer be found. Start over!

Probable causes are storing past the limits of an array, storing a string that is longer than the maximum length of its \$VARIABLE, or using bad limits on MOVE, -MOVE, FILL, BLANK, ERASE, etc.

To prevent the first, check the number of operations vs. size of array, and which value comes first on stack. To prevent the next, force a proper LEFT\$ truncation before storing the string. If the others give you trouble, you can debug by temporarily adding a dummy word instead, such as:
: MOVEX MOVE . . . QUIT ;
Use this to do the operation and to examine what is on stack at that point.

AND OTHERS:

Resourceful Fortb programmers can find other ways to cause these and related troubles. For example:
Mixing up @ and C@, I and CI, etc.
Inserting an inappropriate stack operation between a paired <R and R> .
Running a 32K or 48K RAM-sized program on a smaller-sized MMSFORTH System. You can check available RAM space before and after loading, with 'S PAD - . (numbers larger than 32K will appear negative).

IT COULD BE HARDWARE:

We hope not but it does happen - a particular bug in a RAM chip, poor circuit trace, etc., can look like a bug in one program only. If your program misbehaves identically on another TRS-80 (running at the proper RAM size), it's not a hardware problem.

GET-TOGETHER

Consider sharing your questions and answers with a MMSFORTH User Group, or contact MMS for help in starting one in your metropolitan area. Here is our present list of contacts for local MMSFORTH User Groups:

Morris Herman, 503 Rosario Drive, Santa Barbara CA 93110 (805/964-7144).
Rich Royea, 6456 Lubau, Woodland Hills CA 91367 (213/704-6859).
Ed Laughery, 14317 Royal Oak Avenue, Baton Rouge LA 70816 (504/292-5505).
Jim Gerow, 1630 Worcester Road, Framingham MA 01701 (617/872-1882).
Kim Watt, Box 1013, Berkeley MI 48072 (313/288-9422).
Larry Goforth, 10203-J Golden Meadow, Austin TX 78758 (512/836-0981).
Jim Shepard, 16210 Arbor Downs Drive, Dallas TX 75248 (214/661-9702).
Paul Van der Eijk, 4910 Fran Place #204, Alexandria VA 22312 (703/354-7443).
Rod Proctor, 13520 N.E. 29th Place, Bellevue WA 98005 (206/885-4171 days, 206/883-1923 eves.). Rod also is on THE SOURCE.

NOTE: Program trading is one popular facet of these meetings, but NOT commercial programs and WITHOUT MMSFORTH systems aboard! Promote legitimate sharing, discourage pirating, and take care not to jeopardize your own MMSFORTH serial number.

AT MILLER MICROCOMPUTER SERVICES

WHAT'S COMING:

MMS has delivered many recent development projects at no extra cost, in MMSFORTH V1.9 and DATAHANDLER V1.1. By the next Newsletter issue, we hope to share an Absolute Assembler and real-time interrupt and telephone-dialer programs, the latter illustrating an article on <BUILDS and DOES>.

Our RS-232-C driver and TERMINAL programs are now available on a custom basis, and will be included in a full communications package in the future. This package will permit telephone transfer of Fortb blocks and the operation and debugging of one computer from another - ideal for our consulting work!

Projects concerning advanced files handling, full business systems, and FORTHWRITE (our word-processing program) are progressing well.

HOME CONTROLLER:

Phil Mork, a Weston MA user of Tape MMSFORTH, shares a marvelous project with you. Instead of those \$100-\$200 boards to interface with the Sears (or BSR) Home Controller Systems, Phil uses a \$2 crystal earphone and this free Forth program! The rest of the project will cost about \$40 for the console unit and \$15 for each remote wall unit. Phil warns you NOT to buy your controller unit from Radio Shack and to be careful if selecting one of the BSR models - apparently only the ones from Sears are sure to be compatible with his trick, which is to drive it through the cassette port, "beeping" the appropriate tones into the ultrasonic receiver which is built into some of the console units. If the store can demonstrate a remote ultrasonic unit, you know its BSR console unit is the proper variety.

The crystal earphone is slightly uncommon, but produces better ultrasonic output than does the garden variety. The hardware interfacing is easy as pie! Plug the earphone into the EAR jack on your cassette recorder, then hold it against the ultrasonic detector (just below the red light on the front of the console unit) with a rubber band. Put the recorder in Record mode, remove its motor remote control cable, leave the computer's output cable in the recorder AUX jack, and you're interfaced! Your CRT radiates a similar signal, so for best effect keep several feet away from it or acoustically shield the controller.

Phil has provided several Forth words to control the home controller. Then he uses them to sequence some light bulbs remotely. Using the same ideas and your imagination, you can have fun all over your home or can control more serious equipment in your working environment. Please share your best applications with us - and with Phil Mork!

BLOCK : 49

```

0      ( MMSFORTH DRIVER FOR SEARS HOME CONTROL SYSTEM )
1      ( TWO SCREENS BY PHIL MORK, WESTON, MASS., 4/80 )
2      ( CONNECT XTAL EARPHONE TO GRAY CASSETTE PLUG )
3      ( FASTEN EARPHONE TO CONSOLE BELOW LED )
4      ( USE SEND WITH CONTROL CODE ON STACK )
5 : TASK ;
6 BASE C@ HEX
7 CODE TONE DE POP HL POP BEGIN BEGIN D A MOV FF OUT E A MOV
8      FF OUT L DCR =0 END H DCR =0 END NEXT
9 : 1BIT 19C 201 TONE 1B1 0 TONE ;
10 : 0BIT 130 201 TONE 21D 0 TONE ;
11 : SBIT 2F4 201 TONE 1A7 0 TONE ;
12 : FIVE 5 0 DO 2* DUP 20 AND IF 1BIT ELSE 0BIT THEN
13      LOOP DROP ;
14 : SEND DUP 1BIT FIVE MINUS 1 - FIVE SBIT ;
15 BASE C!

```

BLOCK : 50

```

0      ( DRIVERS FOR SEND, HOME CONTROLLER, P.MORK )
1
2 BASE C@ DECIMAL
3 : WAIT 3000 0 DO LOOP ; : SENDW SEND WAIT ;
4 : 1SEL 12 SENDW ; : 2SEL 28 SENDW ; : 3SEL 4 SENDW ;
5 : 4SEL 20 SENDW ; : 5SEL 2 SENDW ; : 6SEL 18 SENDW ;
6 : 7SEL 10 SENDW ; : 8SEL 26 SENDW ; : 9SEL 14 SENDW ;
7
8
9 : ON 5 SENDW ; : OFF 7 SENDW ; : LIGHTSON 3 SENDW ;
10 : ALLOFF 1 SENDW ; : BRIGHT 0 DO 11 SEND LOOP WAIT ;
11 : DIM 0 DO 9 SEND LOOP WAIT ;
12
13
14
15 BASE C!

```

DISK DRIVES:

We have discovered a nasty bug which infects a very occasional 1771 disk controller cbip. Possibly the 1771 in your TRS-80 clears the data register when indexing through Track 0. This bug manifested itself on one TRS-80 as an apparently missing DATAHANDLER file directory block, which then showed up in fine shape following an ERASE-CORE. The same diskette ran perfectly on other systems. To guard against this unusual and undesirable situation, change MMSFORTH's RAM Location 20878 from a 24 to a 28, as follows:
Remove the write-protect tab and boot your MMSFORTH System. Then:
28 20878 C!
ERASE-CORE 19200 0 0 2 40 DWTSECS .

That will eliminate the problem, and will not adversely affect other operations. MMS recommends this change for all MMSFORTH System Diskettes, Version 1.8 and earlier. Also do the same on precompiled DATAHANDLER and other application diskettes - but remember to increase the "40" (that is, the number of consecutive disk sectors for writing) as required.

Have you adjusted your MMSFORTH System upward from 35-tracks per drive? Our face is red, especially if your disk drives started misbehaving on high-number tracks after you upped the MMSFORTH System capacity to 40 tracks or more in accordance with our early instructions. Near the bottom of Page 2.4 of the MMSFORTH System Instructions Part I, Section 3.4 should read:
40 ' #TRACKS ! 99 22683 C! 39 20916 C! ERASE-CORE 19200 0 0 2 40 DWTSECS . (40 TRACKS)
77 ' #TRACKS ! 192 22683 C! 76 20916 C! ERASE-CORE 19200 0 0 2 40 DWTSECS . (77 TRACKS)

MMSFORTH user Tom White of Laurel, Mississippi gets a free one-year subscription to this Newsletter for first report of this problem. Thanks, Tom! Sorry, folks!

MMSFORTH MODIFICATIONS

NEW RADIO SHACK UPPER-LOWER CASE DISPLAY MOD:

Now that Radio Shack has allowed us to buy back the ASCII code for our lower case character set, we must undistort the driver routines which the Level II ROM and others previously were forced to use. MMSFORTH V1.9 now has a Radio Shack upper-lower-case-mod compatible video driver routine. MMSFORTH always had the lower-case capability; without the appropriate mod in the keyboard unit, lower case text had to be entered in the full-screen editor mode while depressing the shift key. If you've added Radio Shack's new mod, some apparently normal displayed characters will now be coded very low. For an example of this, use a RS ULC-modified keyboard unit and boot MMSFORTH. Immediately after booting, enter:
: TT 64 0 DO 15360 I + C? LOOP ; CR TT

This routine will display the ASCII codes of the first line on the video display, which reads MMSFORTH SYSTEM DISKETTE, etc. Now, we know that ASCII Code 13 is not an M, but our modified system does not! Most prior MMSFORTH code will work OK, but here is a recoding of the ALFA definition used in the JKL function on Block 34 of your pre-V1.9 MMSFORTH System (and also in the ADUMP routine in the March-April MMSFORTH Newsletter). Note that the first half-line remains as before.
: ALFA OVER OVER 0 DO DUP I + C@ DUP 32 < 1F 64 + ELSE DUP 126 > 1F 46 SWAP DROP THEN THEN OVER I + C! LOOP DROP ;

THE LAST WORD: "GO FORTH!"
- from Larry Goforth, a new user who ought to know!

